DECODING MULTIPLE HMM SETS USING A SINGLE SENTENCE GRAMMAR

Field of Invention

[0001] This invention relates to speech recognition and more particularly to a speech recognition search method.

Background of Invention

[0002] Speech recognition devices are typically deployed in different acoustic environments. An acoustic environment refers to a stationary condition in which the speech is produced. For instance, speech signal can be produced by male speakers, female speakers, in office environment, in noisy environment.

[0003] A common way of dealing with multiple environment speech recognition is to train a sets of models such as Hidden Markov Models (HMM) for each environment. Each set of HMMs will have the same number of models representing the same sounds or words spoken in the environment corresponding to the HMM set. Typically, a speech recognizer utilizes a grammar network which specifies the sequence of HMMs that correspond to the particular speech sounds and words making up the allowable sentences. In order to handle the sets of HMMs for each environment, current art technology provides the speech recognizer with a large grammar network which contains a grammar sub-network for each HMM set according to each of the environments. These sub-networks enable the use of each HMM sets within the recognizer. Since the HMM sequences corresponding to sentences allowed by the grammar network generally do not change with environment each grammar sub-network has the same structure. For example, there would be a pronunciation set or network of HMMs (grammars) for male speakers and a set of HMMs for female speakers because the sounds or models for a male speaker are different from a female speaker. At the recognition phase, HMMs of all environments are decoded and the recognition result of the environment giving the maximum likelihood is considered as final results. Such a practice is very efficient in recognition performance. For example, if male/female separate models are not used, with the same amount of HMM parameters, the Word Error Rate (WER) will typically increase 70%.

[0004] More specifically, for a given sentence grammar network, the speech recognizer is required to ~~decode~~ develop high probability paths for M (the number of environments) sub-networks, referencing M sets of HMMs, each of which models a specific acoustic environment. In order to perform acoustic matching with each of the environments, present art recognition search methods typically (which include state-of-the-art recognizers as HTK 2.0) require a grammar network consisting of M sub-networks, as illustrated in Figure 1. Requiring M ~~sets of sentence network~~ sub-networks makes the recognition device more costly and requires much more memory.

Summary of Invention

[0005] A new speech recognition search method is described here, ~~which needs only 1 out of~~ wherein the speech recognizer only requires a grammar network having the size of a generic base sub-network that represents all of the M ~~subnetwork~~ sub-networks ~~(sentence network)~~ and yet gives the same recognition performance, thus reducing the memory requirement for grammar network storage by (M-1) /M. The speech recognition method includes a ~~basic~~ generic base ~~speaker~~ independent grammar ~~or~~ network ~~(sentence structure)~~ specified using a set of independent base symbols and ~~virtual~~ sets of expanded symbols ~~representing the network of expanded sets HMM sets where the pronunciation of each symbol is specified by a set of HMM states~~ referencing the models of each of environment-dependent models sets such as a male and female set. The new speech recognizer builds recognition paths ~~defined on the expanded~~ by expanding the symbols~~,~~ ~~and accesses~~ of the base grammar network ~~using~~ ~~base-symbols,~~ through proper conversion function that gives for each of the ~~base-symbol~~ base grammar network symbols, an enumeration of an ~~any~~ expanded set of symbols and the dependent models referenced by the expanded symbols, and vice versa. That is, it can provide, for one of the symbols of the expanded set of symbols, the base symbol to which it corresponds.

Description of Drawing:

In the drawing:

[0006] Figure 1 illustrates conventional recognizer~~s~~ grammar network that requires ~~large network~~ multiple sub-networks to recognize multiple ~~HMM~~ environment-dependent model sets;

[0007] Figure 2 illustrates a block diagram of the ~~system~~ speech recognition path probability scoring process according to one embodiment of the present invention; and

[0008] Figure 3 illustrates the main ~~program~~ loop procedure of the speech recognition path probability scoring process.

Description of Preferred Embodiment

[0009] In the present application we refer to a node in the grammar network describing the allowed sentences ~~grammar~~ as a *symbol* which references a particular HMM or a group of M HMMs, one from each of the M HMM sets. For typical recognizers, ~~a symbol has to be duplicated M-times in the network when M-sets of HMM are used~~ when M sets of the HMMs are used , then M sub-networks must be in the grammar network, with M sub-networks corresponding to the M environments .This is illustrated in Figure 1, where for three HMM sets there are three sub-networks of nodes ~~of sentence networks are depicted~~.

[0010] In accordance with the present invention, a generic base grammar network is constructed to represent a merged version of the M networks that is speaker independent. For the male and female case this would be a merged version of the male and female networks and be gender-independent. The models for children may also be merged. Other environments may also be merged. We need to further decode specific HMMs such as those for ~~both~~ the male, female and child and combine with the generic base grammar (speaker independent) network where for ~~both~~ male, female and child have the same nodes and transitions.

[0011] In applicant's method of decoding M HMM sets, two types of symbols are distinguished:

- *Base-symbols (α):* Symbols representing the nodes of the ~~basic~~ generic base grammar ~~or~~ network (i.e., the network before duplication for M-sets HMM). To each of the base-symbols there correspond M expanded-symbols which represent the symbols of conceptual sub-networks. They (base symbols and base network) have physical memory space for storage. This is generic (speaker independent) representing the nodes and transitions.

- *Expanded*-symbols (*ã*): Symbols representing the ~~network of M-1~~ expanded grammar network nodes that reference the HMMs from each HMM sets~~.~~ The expanded-symbols are used to construct a conceptual expanded grammar network that simulates the characteristics of an M sub-network grammar. Their existence in the grammar network is conceptual and does not require storage for symbol information, grammar nodes or transitions. ~~This symbol may represents for example the two sets of HMMs for male and female.~~ The expanded-symbols may reference, for example, HMMs from the male and female sets.

[0012] For each base-symbol in the base grammar network there are M ~~-1~~ corresponding expanded-symbols ~~associated~~. The new recognizer builds recognition paths defined on the expanded-symbols, and accesses the network using base-symbols, through proper conversion function that gives the base-symbol of any expanded symbols~~.~~ or the expanded–symbols for a given base-symbol.

[0013] Referring to Figure 2 there is illustrated the ~~system~~ speech recognition path construction and path probability scoring process according to one embodiment of the present invention. For the male and female combined case the generic base grammar network represented by the base symbols α is stored in memory 21. This provides the network structure itself. Also stored in memory 23 is a set of HMMs for male and a set for female for example. A set of HMMs may also be for child. As is well known in the art, speech recognizers process short sequential portions of speech, referred to as frames. Also known in the art, for each incoming speech frame, the speech recognizer must determine which HMMs should be used to construct high probability paths through the grammer network. The base symbol contains the sentence structure. The process is to identify the HMM to be used. For every incoming speech frame a main loop program performs a recognition path construction and update-observation-probability. The main loop program (see Figure 3) includes a path–propagation program 25 and an update-observation-probability program 27. The speech path construction and the scoring process 25 uses the base grammar network and base-symbol information in memory 21 , the HMM model set information in memory 23, and conversion function to identify the expanded symbols to be used to construct the recognition paths through the conceptual expanded network. After the path construction is accomplished the recognizer updates the observation probability 27 of each of the conceptual expanded grammar network paths.

[0014] The function MAIN-LOOP program illustrated in Figure 3 performs recognition path construction for every incoming speech frame:

```
MAIN-LOOP (networks, ~~models~~ hmms):
Begin
    For t = 1 to N Do
    Begin
        PATH-PROPAGATION (network, ~~models~~ hmms, t):
        UPDATE-OBSERVATION-PROB (network, ~~models~~ hmms, t);
    End
End
```

where t indicates the time index of each speech frame, N is the number of frames in the spoken utterance, network represents the generic base grammar network, and hmms represents an ordered listing of all HMMs in the M HMM sets.

[0015] The MAIN_LOOP procedure illustrated in Figure 3 performs recognition path construction for each incoming speech frame. The MAIN_LOOP procedure includes the path construction (Figure 2,25) and the update-observation-probability procedure (Figure 2, 27). After the speech recognizer performs the MAIN-LOOP procedure for all utterance frames, the recognizer then selects the single remaining path that had the highest final probability as the path containing the recognized utterance.

[0016] ~~A path consists of a sequence of symbols, and the pronunciation of each symbol is specified by a set of hidden Markov model states.~~ Since speech frames associated with paths through each expanded-symbol are further associated with a sequence of HMM states of the HMM referenced by the expanded-symbol, paths through the conceptual expanded grammar network consist of a sequence of both expanded-symbols and HMM states. Consequently, for a given utterance frame a path can be extended either within the presently active HMM referred by an active expanded-symbol using within-model–path or another expanded-symbol and its referenced HMM using a cross-model-path, which the decoding procedure constructs for each symbol:

```
PATH-PROPAGATION (network, hmms, t):
Begin
    For each active $\tilde{a}$ at frame $t - 1$ Do
    Begin
        $(\Delta_{hmm,}\ \Delta_{sym}, \alpha)$ = get-offsets ($\tilde{a}$, network);
        hmm = hmms[hmm-code(symbol-list(network)[$\alpha$]) + $\Delta_{hmm,}$];
        WITHIN-MODEL-PATH (hmm, . $p^{\tilde{a}}_{t-1}$, $p^{\tilde{a}}_{t}$);
        CROSS-MODEL-PATH (hmms, network, $\tilde{a}$, $\alpha$, $\Delta_{hmm,}$ $\Delta_{sym}$, t, score ($p^{\tilde{a}}_{t-1}$,
        EXIT-STATE));
    End
End
```

where:

- p^{s}_{t} denotes the storage of path information for the expanded-symbol s at frame t.
- "get-offsets" gives the offset of HMM (Δ_{hmm}), offset of symbol (Δ_{sym}) and the base-symbol (α), given $\tilde{a}$ and a network.
- "symbol-list" returns the list of symbols of a network.
- "hmm-code" gives index of an hmm, associated to a symbol.
- Score (p, i) gives the score at state i of the symbol storage p. We keep what is the symbol and frame from which we are from t to t-1 and trace the sequence of the word . The nodes are constructed based on the model.

[0017] The PATH-PROPAGATION procedure first determines the HMM and base-symbol corresponding to each active expanded-symbol. In order to determine the HMM corresponding to each active expanded-symbol, the recognizer uses a conversion function, ("get offsets"), which provides a parameter (Δ_{hmm}) which can be used to determine the HMM corresponding to the expanded-symbol, and also determines the generic base grammar network base-symbol α in 21 corresponding to the extended-symbol. These are used to access tables consisting of a base-symbol table (symbol_list (network)), an Hmm table for each HMM set and an ordered hmm, via the base symbol α to determine the group of extended-symbols corresponding to the base-symbol. Finally, the parameter (Δ_{hmm}) is used to access a list of HMMs to retrieve the HMM corresponding to the active expanded-symbol. In the search algorithm for each frame time interval 1 to N for frame time t looks back at time t-1 and calculates to find out the base symbol. See Figure 2. From this to access the generic network 21 given the expanded symbol $\tilde{a}$ to get the offset of HMM (Δ_{hmm}). Once the ΔHMM is determined, the HMM memory 23 can be accessed such that the HMM that corresponds to the male base or female is provided. Once the HMM is obtained the sequence of states within model path is determined and then the cross model path. The sequence of HMM states is constructed in the recognition path construction 25 in both the within HMM path and the between models. There are therefore two key functions for decoding, within-model-path construction and cross-model-path construction. The PATH-PROPAGATION procedure then extends the path of the active expanded-symbol with the referenced using th WITHIN-MODEL-PROCEDURE :

```
        WITHIN-MODEL-PATH (hmm, $p_{t-1}$, $p_t$);
        Begin
                For each HMM state $i$ of hmm Do
                Begin
                For each HMM state $j$ of hmm Do
                Begin
                        score ($p_t$, $j$) = score ($p_{t-1}$, $i$) + $a_{i,j}$;
                        from-frame ($p_t$, $j$) = from-frame ($p_{t-1}$, $i$)
                        from-symbol ($p_t$, $j$) = from-symbol ($p_{t-1}$, $i$)
                End
        End
End
```

where:

- $\alpha_{i,j}$ is the transition probability from state i to state j. "from-frame" expands the frame path information in p and "from-symbol" expands the symbol path information in p. The WITHIN-MODEL PATH procedure determines which states of an active HMM can be extended from frame t-1 to frame t, and then extends the path information in p_{t1}- storing the extended information in pt corresponding to $\tilde{a}$.

[0018] When we do the within HMM path, we need to do the storage of t and t-1. That sentence with the highest score is determined based on the highest transition log probability. This is done for every state in the HMM. (For each state j in the equation below. Once we arrive at the end we go back and find out what is the sequence of the symbols that has been recognized. This is stored.

[0019] After PATH-PROPAGATION extends the paths within extended-symbol HMMs, it determines if the path can be extended to other HMMs references by other expanded-symbols using the CROSS-MODEL-PATH procedure.

```
[0020]     CROSS-MODEL-PATH (hmms, network, $\tilde{a}$, $\alpha$ $\Delta_{hmm}$, $\Delta_{sym}$, t, $\delta i$);
Begin
    For each next symbol $s$ of $\alpha$Do
    Begin
        hmm = hmms [hmm-code(symbol-list(network)[s]) + $\Delta_{hmm}$];
        For each HMM initial state $j$ of hmm Do
        Begin
            score $(p^{\tilde{e}}, j) = \delta i \times \pi(j)$;
            from-frame $(p^{\tilde{e}}, j) = t - 1$;
            from-symbol $(p^{\tilde{e}}, j) = \tilde{a}$;
        End
    End
End
```

where "hmm" is the HMM referenced by the expanded-symbol $\tilde{e}$ coming from the same HMM set as expanded-symbol $\tilde{a}$ corresponding to the base-symbol e; δ is the path probability of the path exiting the HMM referenced by expanded-symbol $\tilde{a}$; π $(j)$is the entry probability of HMM "hmm" state j; and $p^{\tilde{e}}$ represents the path storage information for the expanded-symbol $\tilde{e}$.

[0021] CROSS-MODEL-PATH extends the path from the HMM referenced by expanded-symbol $\tilde{a}$ to each HMM referenced by expanded-symbol $\tilde{e}$ subject to the constraint of the base grammar network allowing the expansion to the new HMM.. Note that the CROSS-Model-PATH procedure ensures that extension of the path will only occur within an HMM set, via the Δ_{hmm} parameter and a set of tables consisting of the base-symbol table, an hmm table for each HMM set and an ordered hmm list, so that the recognizer correctly conceptually separates the paths to simulate the M sub-networks.

[0022] For the cross model path we need for the next symbol s of α we need to consider all possible next symbols s. This is the true symbol s (knowledge of grammar that tells which symbol follows which symbol). We determine it's initial state or first HMM and we perform the sequence of HMM states for between states and add the transition probability (log probability) from one state to another. We use the π symbol for outside the states. We go back to the beginning and determine what is the symbol and frame from which we are from so that at the end we can go back and check the sequence of words. By doing this within and between we have constructed all the nodes.

[0023] Finally, once the MAIN-LOOP completes path construction for each of the HMM sets according to the base ~~a path is expanded according to~~ the grammar network, it's path acoustic likelihood score is evaluated by the UPDATE-OBSERVATION-PROB procedure (Figure2,27) :

```
UPDATE-OBSERVATION-PROB (network, models, t);
Begin
        For each active $\tilde{a}$ at frame $t$ Do
        Begin
                ($\Delta_{hmm,}$ $\alpha$) = get-true-symbol ($\tilde{a}$, network);
                hmm = hmms[hmm-code(symbol-list(network)[$\alpha$]) + $\Delta_{hmm}$];
                For each HMM state $j$ of hmm Do
                Begin
                        Evaluate score ( $p_t^{\tilde{a}}$ , $j$);
                End
                calculate score for $\tilde{a}$;
        End
End
```

where:

- "get-true-symbol" returns the base-symbol of a expanded symbol.

[0024] In UPDATE-OBSERVATION-PROB the acoustic likelihood is calculated for each state of each HMM for which there is present path information p_t. The acoustic likelihood is determined by evaluating the likelihood of the present frame of acoustic given the model information of the HMM state. These are all based on the model. The next step is to look at the speech to validate by comparison with the actual speech. This is done in the update-observation-probability program 27. See Figure 2. We need to find the HMM and for every HMM state we need to evaluate the score against the storage area at the time for the symbol α. The highest score is used. The best score models are provided.

RESULTS

[0025] This new method has been very effective at reducing the memory size. Below represents the generic grammar for 1-7 digit strings:

$digit = (zero | oh | one | two | three | four | five | six | seven | eight | nine)[sil];

$DIG = $digit [$digit [$digit [$digit [$digit [$digit [$digit]]]]]];

$SENT = [sil] $DIG [sil];

[0026] It says for we recognize zero or oh or one, or two etc. .It also says a digit is composed of a single digit, two digits etc.. It also says a sentence is on two etc. digits.

[0027] The grammar for the 1-7 digit strings for the old gender dependent way follows:

```
$digit_m = (zero_m | oh_m | one_m | two_m | three_m | four_m | five_m | six_m | seven_m | eight_m | nine_m)[sil_m];
$DIG = $digit_m [ $digit_m [ $digit_m [ $digit_m [ $digit_m [ $digit_m [ $digit_m] ] ] ] ] ];
$SENT_m = [sil_m] $DIG_m [sil_m];
$digit_f = (zero_f | oh_f | one_f | two_f | three_f | four_f | five_f | six_f | seven_f | eight_f | nine_f)[sil_f];
$DIG_f = $digit_f [ $digit_f [ $digit_f [ $digit_f [ $digit_f [ $digit_f [ $digit_f] ] ] ] ] ];
$SENT_f = [sil_f] $DIG_f [sil_f];
$S = $SENT_m | $SENT_f;
```

[0028] This is twice the size of the generic grammar.

[0029] The purpose is to calibrate resource requirement and verify that the recognition scores are bit-exact with multiple grammar decoder. Tests are based on ten files, 5 male 5 female.

[0030] For the grammars above , respectively, a single network grammar of sentence and a multiple (two, one for male, one for female) network grammar of sentence.

COMPUTATION REQUIREMENT

[0031] Due to the conversion between base and expanded symbols, the search method is certainly more complex than the one requiring M-set of networks. To determine how much more computation is needed for the sentence network memory saving, the CPU cycles of top 20 functions are counted, and In are shown in Table 1(excluding three file I/O functions). It can be seen that the cycle:

Item	multiple-grammar	Single-grammar
allocate_back_cell	1603752	1603752
coloring_beam	1225323	1225323
coloring_pending_states *	2263190	2560475
compact_beam_cells	2390449	2390449
cross_model_path *	2669081	2847944
fetch_back_cell	10396389	10396389
find_beam_index	7880086	7880086
get_back_cell	735328	735328
init_beam_list	700060	700060

logGaussPdf_decode	19930695	19930695
log_gauss_mixture	2695988	2695988
mark_cells	13794636	13794636
next_cell	898603	898603
path_propagation *	1470878	1949576
Setconst	822822	822822
update_obs_prob *	5231532	5513276
within_model_path	3406688	3406688

Table 1 CPU cycle comparison for top time-consuming functions (UltraSPARC-II). '*' is introduced to indicate an item that has a changed value.

[0032] Consumption for most functions stays the same. Only four functions showed slight changes. Table 2 summarizes cycle consumption and memory usage. The 1.58% is spent on calculating the set-index, and can be further reduced by storing the indices. However, the percent increase is so law that at this time it might not be worth-doing to investigate the other alternative – CPU efficient implementation.

Item	multiple-grammar	single-grammar	increase
TOP CYCLES	78115500	79352090	1.58
NETWORK SIZE	11728	5853	-50.0

Table 2 Comparison of multiple-grammar vs. single-grammar (memory-efficient implementation).

Abstract

For a given sentence grammar, speech recognizers are often required to decode M sets of HMMs each of which models a specific acoustic environment. In order to match input acoustic observations to each of the environments, typically recognition search methods require a network of M sub-networks. A new speech recognition search method is described here, which needs that is only ~~1 out of the M~~ the size of a single sub-network and yet gives the same recognition performance, thus reducing memory requirement for network storage by (M-1)/M.